## Remarks

This REPLY is in response to the Office Action mailed February 21, 2008. The fee for filing of a Terminal Disclaimer is enclosed herewith. No additional fee is due with this communication.

### I.    Summary of Examiner's Rejections

In the Office Action mailed February 21, 2008, Examiner objected to Claim 1. Claims 1-21 were provisionally rejected for non-statutory double patenting. Claims 1 and 4-7 were rejected under 35 U.S.C. 101 as being directed to non-statutory subject matter. Claims 2 and 8-21 were rejected under U.S.C. 112, second paragraph, as being indefinite. Claims 1-21 were rejected under 35 U.S.C. 102(b) as being anticipated by McNeely et al. (U.S. Patent Publication No. 2002/0162059, hereafter McNeely).

### II.   Summary of Applicant's Amendment

The present Reply amends Claims 1-2, 8-9, and 15-16, leaving for the Examiner's present consideration Claims 1-21. Reconsideration of the Application, as amended, is respectfully requested.

### III.  Claim Objections

Claim 1 was objected to for various informalities. Accordingly, Claim 1 has been amended as shown above. Reconsideration thereof is respectfully requested.

### IV.   Double Patenting

Claims 1-21 were provisionally rejected for non-statutory obviousness type double patenting over the claims of copending Applications 10/814,563 and 10/814,200.

Accordingly, filed together with this REPLY is an appropriate Terminal Disclaimer in compliance with 37 CFR 1.321. Applicant respectfully submits that the filing of a Terminal Disclaimer renders moot the rejection of the claims under the doctrine of obviousness-type double patenting, and reconsideration thereof is respectfully requested.

### V.    Claim Rejections under 35 U.S.C. §101

In the Office Action mailed February 21, 2008, Claims 1 and 4-7 were rejected under 35 U.S.C. 101 as being directed to non-statutory subject matter. Accordingly, Claim 1 has been

amended as shown above. Applicant respectfully submits that Claim 1, as amended, and Claims 4-7, which depend from Claim 1, now conform to the requirements of 35 U.S.C. 101, and reconsideration thereof is respectfully requested.

## VI.     Claim Rejections under 35 U.S.C. §112

In the Office Action mailed February 21, 2008, Claims 2 and 8-21 were rejected under 35 U.S.C. 112, second paragraph, as being indefinite. Accordingly, Claims 2, 8-9 and 15-16 have been amended as shown above. Applicant respectfully submits that the claims now comply with the requirements of 35 U.S.C. 112, and reconsideration thereof is respectfully requested.

## VII.    Claim Rejections under 35 U.S.C. §102

In the Office Action mailed February 21, 2008, Claims 1-21 were rejected under 35 U.S.C. 102(b) as being anticipated by McNeely (U.S. Patent Publication No. 2002/0162059).

## Claim 1

Claim 1 has been amended to more clearly define the embodiment therein. As amended, Claim 1 defines:

*1.      (Currently Amended)  A system that provides a generic user interface testing framework, comprising:*
*a computer including a computer readable medium, and a processor operating thereon;*
*a software application source code, stored on the computer readable medium, wherein the software application source code defines a software application under development, including a graphical user interface as part of the software application, and wherein the software application source code executes on the computer to display its graphical user interface;*
*one or more different software test tools that can be invoked by a user to perform testing operations on the graphical user interface that is displayed while the software application is running, wherein each of the one or more different software test tools understand their own tool-specific scripting language;*
*a test case input file stored on the computer readable medium, that contains a plurality of generic interface commands that are abstractions independent of any tool-specific scripting language, wherein the test case input file can be reused as necessary for testing against a software application's graphical user interface in any of the different software test tools;*
*an interpretive engine that executes on the computer, and that includes a plurality of dynamically loaded libraries corresponding to the plurality of different software test tools, and including a library for each of the one or more different software*

*test tools, wherein the interpretive engine receives the generic interface commands defined in the test case input file, determines which software test tool the user is currently using, loads required libraries to map the generic interface commands to corresponding tool-specific testing operations, uses the software test tool to perform the testing operations on the software application's graphical user interface including translating the generic interface commands to tool-specific commands, and reports to the user the success or failure of the testing operations;*

*an editor that allows the user to edit the test case input file wherein the editor includes a rules-based wizard for assisting the user in generating different generic interface commands; and*

*wherein the rules-based wizard includes a plurality of user interface testing operations and wherein the rules-based wizard guides the user to pick from among the plurality of user interface testing operations to build the different generic interface commands.*

Claim 1 has been amended to more clearly define the embodiment therein as comprising one or more different software test tools that can be invoked by a user to perform testing operations on a graphical user interface, wherein each of the one or more different software test tools understand their own tool-specific scripting language. A test case input file contains a plurality of generic interface commands that are abstractions independent of any tool-specific scripting language. The test case input file can be edited and reused as necessary by the user to specify different generic interface commands for testing against a software application's graphical user interface in any of the different software test tools. An interpretive engine that includes a library for each of the one or more different software test tools, receives the generic interface commands, determines which software test tool the user is currently using, and translates the generic interface commands to tool-specific commands.

The advantages of the embodiment defined by Claim 1 include that the system maps generic interface commands into tool-specific testing operations. While automated test development systems, suites and tools have been developed, the typical approach of such automated test development tools requires that the operator have knowledge not just of the test-tool-specific scripting language and environment, but also the specific features and idioms of the vendor-specific tool environment. The learning curve for these testing tools can be significant, and is often of use solely with that one tool. In accordance with the embodiment of Claim 1, the system insulates testers from the tool-specific testing operations of a particular testing tool, so that effort spent testing the build of a software application can be reduced.

McNeely discloses a communications network test system that facilitates autonomous or attendant-free interaction between the administrative interfaces of multiple network devices under test. The test system includes device-specific communication interface packages that

map generic commands to device-specific commands. (Abstract). Each device-specific communication interface package contains information for mapping executable instructions between a common or abstract command language and the command line interface language of a specific device. (Paragraph [0033]). Devices under test may include, but are not limited to: a service control point (SCP) **218**, a home location register (HLR) **219**, a visitor location register (VLR) **220**, a short message service (SMSC) **221**, a domain name system (DNS) server **222**, a data packet router **224**, a signaling gateway (SG) **226** , a network simulator and monitoring device **228**, a signal transfer point **229**, and a softswitch (SS) node **230**. (Paragraph [0038]).

Claim 1, in contrast to McNeely, includes a test case input file stored on the computer readable medium, that contains a plurality of generic interface commands. This test case input file can be used with a variety of test software tools. McNeely, as described above, appears to be a test tool for testing a network including a variety of network devices. Because each device may not respond to the same commands, McNeely appears to describe a system in which generic commands can be used to test the variety of network devices. The system translates the generic commands into device-specific commands during the test. By using generic commands a user can create a comprehensive series of tests that test the various devices that exist in a given network without needing to know the specific commands of each device.

These generic commands, however, while device-generic do not appear to be tool-generic. For example, if a user writes a test file using McNeely's generic commands but then chooses to use a competitor's test tool, it does not appear that the user can use the same test file, without modification. Thus, the commands appear to remain tool-specific. However, in the embodiment of Claim 1, a test case input file contains a plurality of *generic interface commands that are abstractions independent of any tool-specific scripting language*. The test case input file can be reused as necessary by the user to specify different generic interface commands for testing against a software application's graphical user interface in *any of the different software test tools*. Applicant respectfully submits that McNeely does not disclose or render obvious these features.

In view of the above comments, Applicant respectfully submits that Claim 1, as currently amended, is neither anticipated by nor obvious in view of the cited references, and reconsideration thereof is respectfully requested.

**Claims 2-7, 9-14, and 16-21**

Claims 2-7, 9-14, and 16-21 have not been addressed separately but it is respectfully submitted that these claims are allowable as depending from an allowable independent claim, and further in view of the comments provided above. Applicant respectfully submits that Claims 2-7, 9-14, and 16-21, are similarly neither anticipated by, nor obvious in view of the cited references and reconsideration thereof is respectfully requested.

**Claims 8 and 15**

The comments provided above with respect to Claim 1 are hereby incorporated by reference. Claims 8 and 15 have been similarly amended to more clearly define the embodiments therein. For similar reasons as provided above with respect to Claim 1, Applicant respectfully submits that Claims 8 and 15, as amended, are likewise neither anticipated by, nor obvious in view of the cited references, and reconsideration thereof is respectfully requested.

**VIII.    Conclusion**

In light of the above, it is respectfully submitted that all of the claims now pending in the subject patent application should be allowable, and reconsideration thereof is respectfully requested. The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,


Date:  May 21, 2008                           By:    /Nathan L. Feld/
                                                     Nathan L. Feld
                                                     Reg. No. 59,725


FLIESLER MEYER LLP
650 California Street, 14th Floor
San Francisco, California  94108
Telephone: (415) 362-3800
Customer No. 23910